AMENDMENTS TO THE CLAIMS:

1. (Currently amended) A method of improving at least one of speed and efficiency when executing a level 3 dense linear algebra ~~subroutine~~ processing on a computer, said method comprising:

automatically setting an optimal machine state on said computer for said processing by selecting an optimal matrix subroutine from among a plurality of matrix subroutines stored in a memory that ~~performs~~ could alternatively perform a level 3 matrix multiplication processing.

2. (Original) The method of claim 1, wherein said computer includes an L1 cache, said method further comprising:

determining a size of each of matrices involved in said matrix multiplication; and

selecting one of said matrices to reside in an L1 cache, based on said determined size,

wherein said selecting a matrix subroutine comprises determining which of said matrix subroutines is consistent with said matrix selected to reside in said L1 cache .

3. (Currently amended) The method of claim 1, wherein said matrix subroutine comprises a substitute of a subroutine from ~~a~~ LAPACK (Linear Algebra PACKage).

4. (Currently amended) The method of claim 3, wherein said substitute LAPACK subroutine comprises a Basic Linear Algebra Subroutine (BLAS) Level 3 L1 cache kernel.

5. (Currently amended) The method of claim 1, wherein said selecting a matrix subroutine comprises an aspect of a generalized matrix streaming process in which matrix data is stored in multiple levels of computer memory, including a matrix block stored in an L1 cache and matrix data of two other matrices stored in at least one higher level of cache, such that ~~and~~ said matrix data of said two other matrices is systematically streamed into said matrix multiplication processing through said L1 cache.

6. (Currently amended) The method of claim 1, wherein said plurality of matrix subroutines comprises six possible matrix subroutines that could alternatively be used for said level 3 matrix multiplication processing.

7. (Currently amended) An apparatus, comprising:

a memory to store matrix data to be used for a processing in a level 3 dense linear algebra program;

a processor to perform said processing; and

a selector to select an optimal one of a plurality of possible matrix subroutines ~~to~~ that could alternatively perform said processing, thereby automatically setting said apparatus into an optimal machine state to perform said processing.

8. (Currently amended) The apparatus of claim 7, further comprising an L1 cache, wherein said selector makes the selection by:

determining a size of each of matrices involved in said ~~matrix multiplication~~ level 3 processing; and

selecting one of said matrices to reside in said L1 cache, based on said determined sizes,

wherein said selecting a matrix subroutine comprises determining which of said matrix subroutines is consistent with said matrix selected to reside in said L1 cache.

9. (Currently amended) The apparatus of claim 7, wherein said matrix subroutine comprises a substitute of a subroutine from a LAPACK (Linear Algebra PACKage).

10. (Currently amended) The apparatus of claim 9, wherein said substitute LAPACK subroutine comprises a Basic Linear Algebra Subroutine (BLAS) Level 3 L1 cache kernel.

11. (Original) The apparatus of claim 7, wherein said selector for selecting a matrix subroutine includes a storage for storing matrix data in multiple levels of computer memory and a mechanism for streaming said matrix data into said matrix multiplication process.

12. (Original) The apparatus of claim 7, wherein said plurality of matrix subroutines comprises six possible matrix subroutine kernel types.

13. (Currently amended) A ~~signal-bearing~~ machine- readable storage medium tangibly embodying a program of machine-readable instructions executable by a digital processing apparatus to perform a method of improving at least one of speed and efficiency when executing a linear algebra subroutine on a computer, said method comprising:

selecting ~~a~~ an optimal matrix subroutine from among a plurality of matrix subroutines that ~~performs~~ can alternatively perform a level 3 matrix multiplication processing, thereby automatically setting said computer into an optimal machine state for performing said level 3 matrix multiplication processing.

14. (Currently amended) The ~~signal-bearing~~ machine-readable storage medium of claim 13, wherein said digital processing apparatus includes an L1 cache, said method further comprising:

determining a size of each of matrices involved in said matrix multiplication processing; and

selecting one of said matrices to reside in an L1 cache, based on said determined size,

wherein said selecting a matrix subroutine comprises determining which of said matrix subroutines is consistent with said matrix selected to reside in said L1 cache.

15. (Currently amended) The ~~signal-bearing~~ machine-readable storage medium of claim 13, wherein said matrix subroutine comprises a substitute for a subroutine from ~~a~~ LAPACK (Linear Algebra PACKage).

16. (Currently amended) The ~~signal-bearing~~ machine-readable storage medium of claim 15, wherein said substitute LAPACK subroutine comprises a Basic Linear Algebra Subroutine (BLAS) Level 3 L1 cache kernel.

17. (Currently amended) The ~~signal-bearing~~ machine-readable storage medium of claim 13, wherein said selecting a matrix subroutine comprises an aspect of a generalized matrix streaming process in which matrix data is stored in multiple levels of computer memory, including a matrix block stored in an L1 cache and matrix data of two other matrices stored in at least one higher level of cache or other memory, such that ~~and~~ said matrix data of said two other matrices is systematically streamed into said matrix multiplication processing through said L1 cache.

18. (Currently amended) The ~~signal-bearing~~ machine-readable storage medium of claim 13, wherein said plurality of matrix subroutines comprises six possible kernel type subroutines.

19. (Currently amended) A method of providing a service involving at least one of solving and applying a scientific/engineering problem, said method comprising at least one of:

using a linear algebra software package that improves at least one of speed and efficiency to performs one or more matrix processing operations, wherein said linear algebra software package ~~selects a~~ achieves the improved speed or efficiency by selecting an optimal matrix subroutine from among a plurality of matrix subroutines that ~~performs~~ alternatively can perform a matrix multiplication processing, thereby automatically setting a computer into an optimal machine state for performing said matrix multiplication processing;

providing a consultation for solving a scientific/engineering problem using said linear algebra software package;

transmitting a result of said linear algebra software package on at least one of a network, a signal-bearing medium containing machine-readable data representing said result, and a printed version representing said result; and

receiving a result of said linear algebra software package on at least one of a network, a signal-bearing medium containing machine-readable data representing said result, and a printed version representing said result.

20. (Currently amended) The method of claim 19, wherein said matrix subroutine comprises a Basic Linear Algebra Subroutine (BLAS) Level 3 L1 cache kernel from a LAPACK (Linear Algebra PACKage).